

CLAIMS:

What is claimed is:

1. A method for monitoring performance of a program being executed using symmetric multiprocessing (SMP) functionality comprising:
 - executing a native code routine;
 - executing a first thread of the native code routine on a first symmetric multiprocessing (SMP) processor;
 - ascertaining first profile information;
 - updating first thread profile information with the first profile information;
 - executing the first thread of the native code routine on a second SMP processor;
 - ascertaining second profile information; and
 - updating first thread profile information with the second profile information.
2. The method recited in claim 1 above, further comprises:
 - executing a second thread of the native code routine on the first SMP processor;
 - ascertaining third profile information; and
 - updating second thread profile information with the third profile information.
3. The method recited in claim 1 above wherein the first profile information relates to the execution of the first thread on the first SMP processor.
4. The method recited in claim 1 above, wherein the first profile information relates to the execution of the

009020 0422960

first thread on a virtual machine.

5. The method recited in claim 1 above, wherein ascertaining first profile information further comprises:
retrieving processor accumulated profile information;

retrieving processor last profile information; and
calculating the first profile information by
comparing the processor accumulated profile information
and the processor last profile information.

6. The method recited in claim 5 above further comprises:

resetting the processor last profile information by
replacing the last processor profile information with the
processor accumulated profile information.

7. The method recited in claim 1 above, wherein ascertaining first profile information further comprises:

accessing a first processor data area containing
first processor accumulated profile information; and
updating the first processor accumulated profile
information with virtual machine profile information,
wherein the first profile information is calculated from
the first processor accumulated profile information.

8. The method recited in claim 7 above, wherein
updating the first processor accumulated profile
information with virtual machine profile information is
performed by a virtual machine.

9. The method recited in claim 7 above, wherein

009020 0422960

updating the first processor accumulated profile information with virtual machine profile information is performed by a virtual machine further comprises:

receiving a request to update the first processor accumulated profile information with virtual machine profile information;

accessing a first processor data area containing first processor accumulated profile information; and

updating the first processor accumulated profile information with virtual machine profile information, wherein updating the first processor accumulated profile information with virtual machine profile information is performed by an operating system kernel.

10. The method recited in claim 1 above, wherein ascertaining first profile information further comprises:

receiving virtual machine profile information;

updating processor accumulated profile information with virtual machine profile information;

retrieving processor last profile information; and

calculating the first profile information by comparing the processor accumulated profile information and the processor last profile information.

11. The method recited in claim 7 above, wherein the first processor data area is a predefined data area.

12. The method recited in claim 7 above, wherein the first processor data area is allocated for first processor accumulated profile information at initialization.

09612340.070600

13. The method recited in claim 1 above, wherein the first processor accumulated profile comprises one of allocation bytes, allocation objects, time, live object and live bytes.

14. A method for monitoring performance of a program being executed by symmetric multiprocessing (SMP) functionality comprising:

launching a current thread of a native code routine on a first symmetric multiprocessing (SMP) processor;
stopping the last thread on the first SMP processor;
accessing a first SMP processor data area;
retrieving first SMP processor accumulated profile information;

retrieving first SMP processor last accumulated profile information;

ascertaining first profile information from the first SMP processor accumulated profile information and the first SMP processor last accumulated profile information;

updating first thread profile information with the first profile information;

setting first SMP processor last accumulated profile information in the first SMP processor data area equal to the first SMP processor accumulated profile information;
and

starting the first thread on the first SMP processor.

15. A method for monitoring performance of a program being executed by symmetric multiprocessing (SMP) functionality comprising:

009020"04E2T960

receiving a request for current thread profile information:

retrieving SMP processor accumulated profile information from each SMP processor's data area;

retrieving SMP processor last accumulated profile information from each processor's data area;

ascertaining profile information from the SMP processor accumulated profile information and the SMP processor last accumulated profile information for each thread running on a processor;

updating thread profile information with the profile information for each thread running on a processor;

setting each SMP processor last accumulated profile information in the respective SMP processor's data area equal to the SMP processor accumulated profile information for that SMP processor; and

transferring the thread profile information to the requester.

16. A method for monitoring performance of a program being executed by symmetric multiprocessing (SMP) functionality comprising:

receiving thread event information related to a change in the value of thread metric being tracked;

updating a first metric value by an amount equal to the change in the metric for the current thread running on the virtual machine, wherein the metric is contained in thread accumulated profile information; and

updating a second metric value by an amount equal to the change in the metric, wherein the metric is contained in total accumulated metric profile information.

09612340.070600

17. The method recited in claim 16 above further comprises:

receiving a request for a thread accumulated profile information;

calculating a current thread accumulated profile information by decreasing the value of the thread accumulated profile information by a value of a last accumulated thread accumulated profile information;

resetting a first last thread accumulated profile information as the thread accumulated profile information; and

returning the current thread accumulated profile information.

18. The method recited in claim 16 above wherein the thread accumulated profile information relates to the execution of a thread on an symmetric multiprocessing (SMP) enabled system.

19. The method recited in claim 16 above, wherein thread event information relates to the execution of the first thread on a virtual machine.

20. The method recited in claim 16 above, wherein the first processor accumulated profile comprises one of allocation bytes, allocation objects, time, live object and live bytes.

21. A data processing system for monitoring performance of a program being executed by symmetric multiprocessing (SMP) functionality comprising:

executing means for executing a native code routine;

009020 "04E2T960

executing means for executing means for executing a first thread of the native code routine on a first symmetric multiprocessing (SMP) processor;

ascertaining means for ascertaining first profile information;

updating means for updating first thread profile information with the first profile information;

executing means for executing the first thread of the native code routine on a second SMP processor; a second thread of the native code routine; and

ascertaining means for ascertaining second profile information; and

updating means for updating first thread profile information with the second profile information.

22. The system recited in claim 21 above, further comprises:

executing means for executing a second thread of the native code routine on the first SMP processor;

ascertaining means for ascertaining third profile information; and

updating means for updating second thread profile information with the third profile information.

23. The system recited in claim 21 above wherein the first profile information relates to the execution of the first thread on the first SMP processor.

24. The system recited in claim 21 above, wherein the first profile information relates to the execution of the first thread on a virtual machine.

009020 04E2T960

25. The system recited in claim 21 above, wherein the ascertaining means for ascertaining first profile information further comprises:

retrieving means for retrieving processor accumulated profile information;

retrieving means for retrieving processor last profile information; and

calculating means for calculating the first profile information by comparing the processor accumulated profile information and the processor last profile information.

26. The system recited in claim 25 above further comprises:

resetting means for resetting the processor last profile information by replacing the last processor profile information with the processor accumulated profile information.

27. The system recited in claim 21 above, wherein the ascertaining means for ascertaining first profile information further comprises:

accessing means for accessing a first processor data area containing first processor accumulated profile information; and

updating means for updating the first processor accumulated profile information with virtual machine profile information, wherein the first profile information is calculated from the first processor accumulated profile information.

28. The system recited in claim 27 above, wherein the

009070 04227960

updating means for updating the first processor accumulated profile information with virtual machine profile information is performed by a virtual machine.

29. The system recited in claim 27 above, wherein the updating means for updating the first processor accumulated profile information with virtual machine profile information is performed by a virtual machine further comprises:

receiving means for receiving a request to update the first processor accumulated profile information with virtual machine profile information;

accessing means for accessing a first processor data area containing first processor accumulated profile information; and

updating means for updating the first processor accumulated profile information with virtual machine profile information, wherein updating the first processor accumulated profile information with virtual machine profile information is performed by an operating system kernel.

30. The system recited in claim 21 above, wherein the ascertaining means for ascertaining first profile information further comprises:

receiving means for receiving virtual machine profile information;

updating means for updating processor accumulated profile information with virtual machine profile information;

retrieving means for retrieving processor last profile information; and

009020"04E2T960

calculating means for calculating the first profile information by comparing the processor accumulated profile information and the processor last profile information.

31. The system recited in claim 27 above, wherein the first processor data area is a predefined data area.

32. The system recited in claim 27 above, wherein the first processor data area is allocated for first processor accumulated profile information at initialization.

33. The system recited in claim 21 above, wherein the first processor accumulated profile comprises one of allocation bytes, allocation objects, time, live object and live bytes.

34. A computer program product for monitoring performance of a program being executed by symmetric multiprocessing (SMP) functionality comprising:

executing instructions for executing a native code routine;

executing instructions for executing a first thread of the native code routine on a first symmetric multiprocessing (SMP) processor;

ascertaining instructions for ascertaining first profile information;

updating instructions for updating first thread profile information with the first profile information;

executing instructions for executing the first thread of the native code routine on a second SMP

009070"04E2F960

processor;

a second thread of the native code routine; and

ascertaining instructions for ascertaining second profile information; and

updating instructions for updating first thread profile information with the second profile information.

35. The computer program product recited in claim 34 above, further comprises:

executing means for executing a second thread of the native code routine on the first SMP processor;

ascertaining means for ascertaining means for ascertaining third profile information; and

updating means for updating second thread profile information with the third profile information.

36. The computer program product recited in claim 34 above wherein the first profile information relates to the execution of the first thread on the first SMP processor.

37. The computer program product recited in claim 34 above, wherein the first profile information relates to the execution of the first thread on a virtual machine.

38. The computer program product recited in claim 34 above, wherein the ascertaining instructions for ascertaining first profile information further comprises:

retrieving instructions for retrieving processor accumulated profile information;

retrieving instructions for retrieving processor last profile information; and

009020-070600

calculating instructions for calculating the first profile information by comparing the processor accumulated profile information and the processor last profile information.

39. The computer program product recited in claim 38 above further comprises:

resetting instructions for resetting the processor last profile information by replacing the last processor profile information with the processor accumulated profile information.

40. The computer program product recited in claim 34 above, wherein the ascertaining instructions for ascertaining first profile information further comprises:

accessing instructions for accessing a first processor data area containing first processor accumulated profile information; and

updating instructions for updating the first processor accumulated profile information with virtual machine profile information, wherein the first profile information is calculated from the first processor accumulated profile information.

41. The computer program product recited in claim 40 above, wherein the updating instructions for updating the first processor accumulated profile information with virtual machine profile information is performed by a virtual machine.

42. The computer program product recited in claim 40 above, wherein the updating instructions for updating the

09612340 " 070600

first processor accumulated profile information with virtual machine profile information is performed by a virtual machine further comprises:

receiving instructions for receiving a request to update the first processor accumulated profile information with virtual machine profile information;

accessing instructions for accessing a first processor data area containing first processor accumulated profile information; and

updating instructions for updating the first processor accumulated profile information with virtual machine profile information, wherein updating the first processor accumulated profile information with virtual machine profile information is performed by an operating system kernel.

43. The computer program product recited in claim 34 above, wherein the ascertaining instructions for ascertaining first profile information further comprises:

receiving instructions for receiving virtual machine profile information;

updating instructions for updating processor accumulated profile information with virtual machine profile information;

retrieving instructions for retrieving processor last profile information; and

calculating instructions for calculating the first profile information by comparing the processor accumulated profile information and the processor last profile information.

44. The computer program product recited in claim 40

009020-070600

above, wherein the first processor data area is a predefined data area.

45. The computer program product recited in claim 40 above, wherein the first processor data area is allocated for first processor accumulated profile information at initialization.

46. The computer program product recited in claim 34 above, wherein the first processor accumulated profile comprises one of allocation bytes, allocation objects, time, live object and live bytes.

009070"04E2T960
09612340"070600